

Lớp 22C1003

Tên: Trần Huy Châu

MSSV: 22127191

Câu 1:

↳ lý do UDP nhanh hơn TCP là

- Không cần phải thiết lập liên kết giữa client - server
- (~~trễ~~ ~~hiển~~ ~~thời~~)
- Đơn giản: không cần ~~trễ~~ ~~hiển~~ ~~thời~~ chờ đợi kết nối giữa người gửi và người nhận
- Header nhỏ
- Không cần phải kiểm soát tốc độ truyền

Câu 2:

Sẽ lựa chọn giữa stop & wait và Pipeline

+ Về số lượng packet được gửi:

Stop & wait: ~~sender gửi~~ ^{sender gửi} sender gửi 1 gói rồi phải chờ nhận ACK từ receiver.

Pipeline: Sender gửi nhiều gói tin, và chờ đợi nhận lại nhiều ACK cùng lúc

+ Về tốc độ và độ hiệu quả

- Stop & wait: mất nhiều thời gian hơn, không tận dụng băng thông
- Pipeline: Tiết kiệm thời gian, tận dụng băng thông

+ Về kiểm soát lỗi

Stop & wait: sender chờ ACK rồi gửi tiếp các gói tin, nếu 1 time-out thì gửi lại gói tin trước đó

- Pipeline: ~~to~~ Sender cũng nhận ACK từ receiver. Nếu có gói tin bị loss thì gửi lại gói đó (Selective Repeat) hoặc là ~~lại~~ gửi lại một nhóm các gói tin bắt đầu từ gói bị loss (Go-Back-N) + về đề phút tập

- Stop & wait: chờ giao hàng

- Pipeline: phút tập hỏi cho cả sender và receiver. Nếu là hiện Selective repeat thì receiver phải có buffer.

Câu 3: So sánh brief giữa Go-Back-N và Selective Repeat:

+ về cách xử lý nếu có gói tin bị loss/corrupted:

- Go-Back-N: Nếu receiver phát hiện có lỗi ở một gói tin thì receiver sẽ xóa toàn bộ gói tin sau gói tin bị lỗi và yêu cầu sender gửi lại toàn bộ các gói tin từ gói tin bị lỗi.

- Selective Repeat: Nếu ~~sender~~^{receiver} phát hiện có lỗi ~~ở~~^{chỉ} ở một gói tin thì nó sẽ chỉ xóa gói tin bị lỗi đó và yêu cầu sender gửi lại gói tin bị lỗi đó. Nếu số trượt bên ~~sender~~^{receiver} sẽ trượt lên theo số ACK nhận được trước gói tin bị lỗi. Khi time out, sender sẽ gửi lại gói tin bị lỗi.

+ Về buffer

- Go-Back-N: không có buffer

- Selective Repeat: Có buffer

+ Về độ hiệu quả: Selective Repeat hiệu quả hơn Go-Back-N

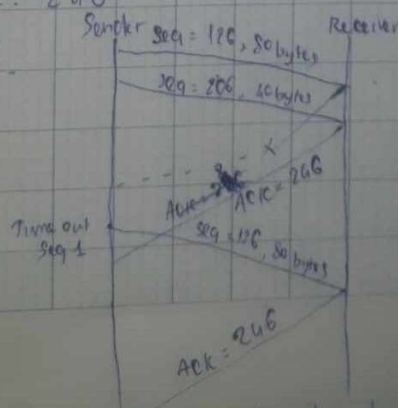
- Gieo thích 2 hình sau:

+ Phần 1: Go-Back-N

- Sender gửi lần lượt các packet 0, 1, 2, 3.
- Sender chỉ nhận được ACK cho gói 0, 1. Gói 2 ^{bị} ~~time out~~
- Sender gửi gói 4, 5 sau khi nhận ACK cho gói 0, 1.
- Sender phát hiện gói 2 bị time out.
- Sender gửi toàn bộ các gói tin chưa biết được của số tuần tự từ gói tin bị time out (gói 2, 3, 4, 5).

- Sender gửi lần lượt các gói 0, 1, 2, 3
- Sender nhận ACK cho gói 0, 1 thì liền gửi gói 4, 5.
- Sender thấy gói 2 bị time out
- Sender gửi lại gói 2
- Receiver nhận được gói 2, 3, 3, 4 (3, 4, 5 lỗi trong buffer)
- gửi lại ACK cho gói 2
- Sender nhận lần lượt ACK từ các gói 3, 4, 5, 2.

1. 206



Câu 1:

- ACK của B gửi tới A là: 126 do receiver chưa nhận được packet 1.
- Khi nhận được B gửi tới ACK = 126 thì sender A sẽ gửi lại packet 1, 80 bytes